QTM 347 Machine Learning

Lecture 22: Introduction to Diffusion Model

Ruoxuan Xiong

EMORY

Diffusion model

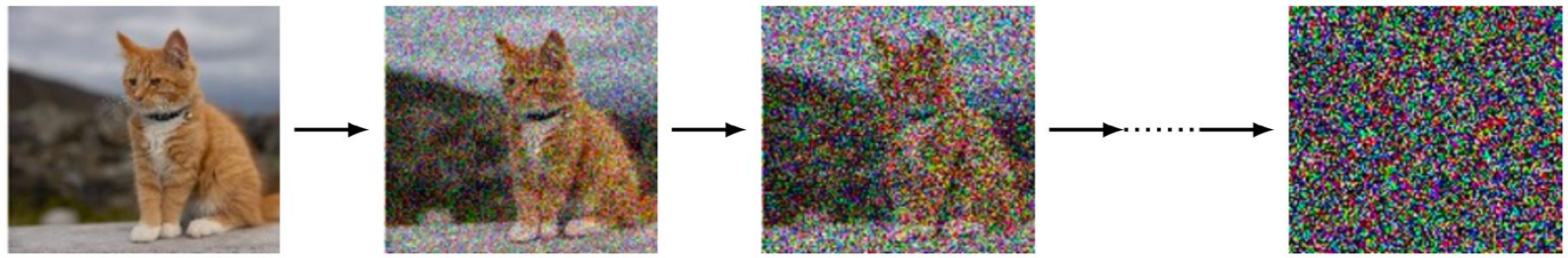
- **Generative models** focus on understanding how the data is generated. They aim to learn the distribution of the data itself
  - LDA and QDA are examples of generative models

- The **diffusion model** is a class of generative models, which has been used in many applications

- In this lecture, for illustration, we focus on models of image data, but the framework has much broader applicability

Central idea of diffusion model I

- The **central idea** is to take each training image and to corrupt it using a multi-step noise process to transform it into a sample from a Gaussian distribution (**forward/noising process**)



  - In the forward process, an image x is gradually corrupted with multiple stages of additive Gaussian noise, giving a sequence of increasingly noisy images

  - After a large number T of steps, the result is indistinguishable from a sample drawn from a Gaussian distribution

Central idea of diffusion model II

- The **forward/noising process** is followed by the **reverse/denoising process**
  - In the reverse process, a deep neural network is then trained to invert this process, and once trained the network can then generate new images starting with samples from a Gaussian as input

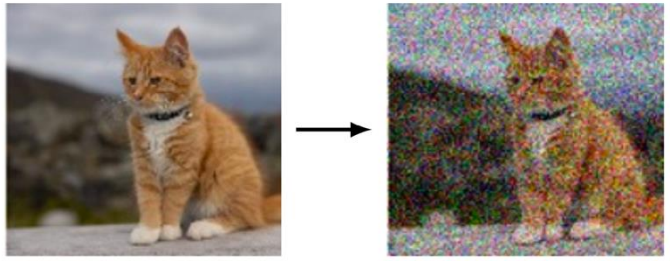
Forward encoder I

- Suppose we take an image from the training data, denoted by x, and blend it with Gaussian noise independently for each pixel to give a noise-corrupted image z_1 defined by

$$z_1 = \sqrt{1-\beta_1}x + \sqrt{\beta_1}\epsilon_1$$

with $\epsilon_1 \sim N(\epsilon_1 \mid 0, I)$ and $\beta_1 < 1$ is the variance of the noise distribution. An alternative representation is

$$q(z_1|x) = N(z_1 \mid \sqrt{1-\beta_1}x, \beta_1 I)$$



$\mathbf{x}$ $\mathbf{z}_1$

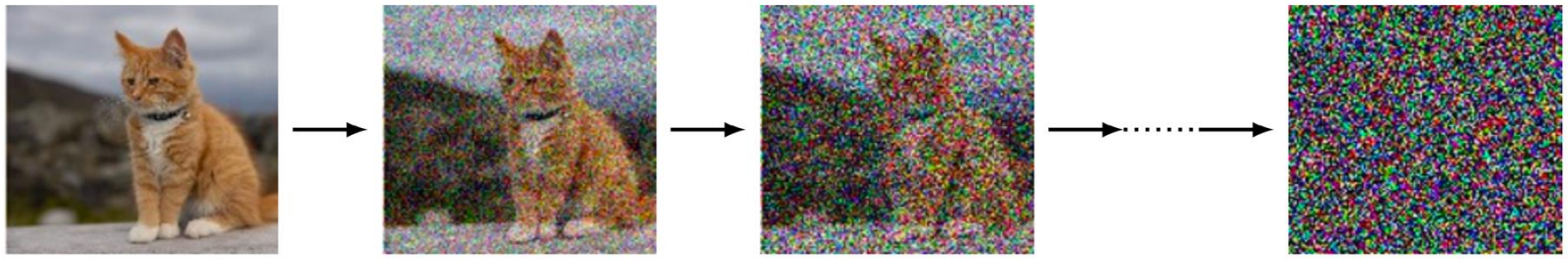
Forward encoder II

- We repeat the process with additional independent Gaussian noise steps to give a sequence of increasingly noisy images $z_2, \cdots, z_T$. Each successive image is given by

$$z_t = \sqrt{1-\beta_t} z_{t-1} + \sqrt{\beta_t}\epsilon_t$$

or equivalently,

$$q(z_t|z_{t-1}) = N(z_t \mid \sqrt{1-\beta_t} z_{t-1}, \beta_t I)$$



Diffusion kernel

- The joint distribution of the latent variables $z_1, z_2, \cdots, z_t$ conditional on the observed data x is given by

$$q(z_1, \cdots, z_t|x) = q(z_1 \mid x) \prod_{\tau=2}^{t} q(z_\tau \mid z_{\tau-1})$$

- We can marginalize over the intermediate variables $z_1, z_2, \cdots, z_{t-1}$, and then obtain the *diffusion kernel*

$$q(z_t|x) = N(z_t \mid \sqrt{\alpha_t}x, (1-\alpha_t)I)$$

where $\alpha_t = \prod_{\tau=1}^{t}(1-\beta_\tau)$

Reverse decoder

- In the reverse decoder, we seek to obtain $q(z_{t-1}|z_t)$. Then we can undo the noise process

- Conceptually, we could use Bayes' theorem

$$q(z_{t-1}|z_t) = \frac{q(z_t \mid z_{t-1})q(z_{t-1})}{q(z_t)}$$

- However, directly estimating $q(z_{t-1}|z_t)$ is intractable!

- This is because we need to estimate $q(z_t) = \int q(z_{t-1} \mid x)p(x)dx$, which requires us to model the unknown data distribution $p(x)$

An alternative solution

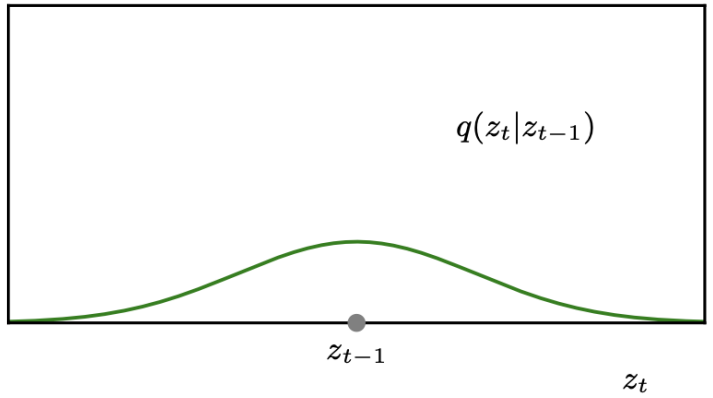
- We instead learn an approximation to the reverse distribution by using a distribution $p(z_{t-1} \mid z_t, w)$ governed by a deep neural network, where w represents the network parameters (e.g., weights and biases)

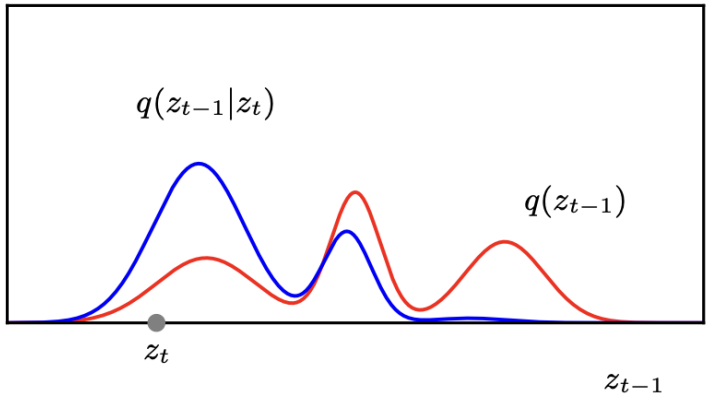
- What would be the distribution of $p(z_{t-1} \mid z_t, w)$?

➢This depends on the variance of noise β_t!

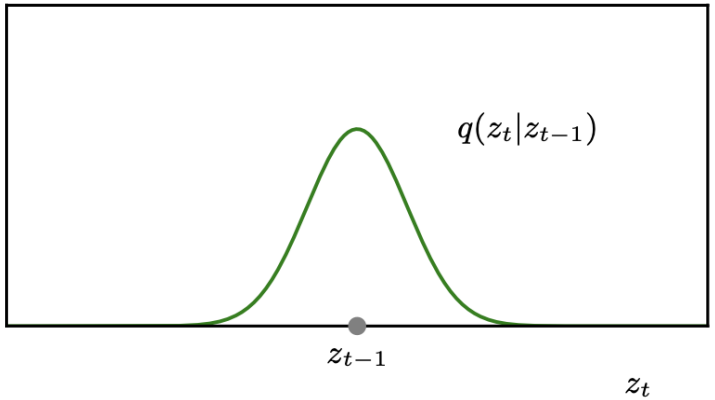
Role of variance of noise

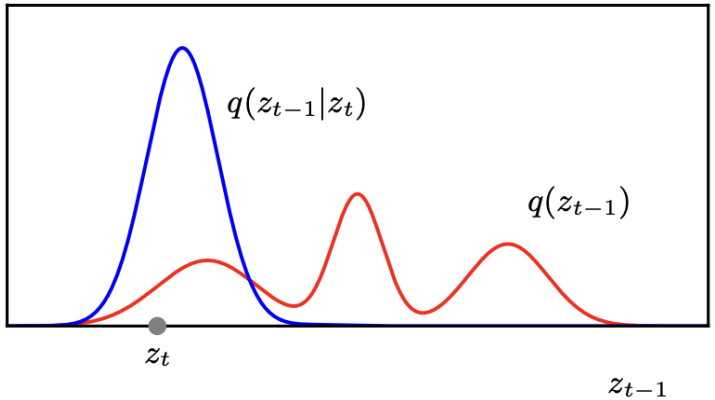
- With a large variance of noise β_t, the distribution $q(z_{t-1}|z_t)$ has a complex multimodal structure





- With a small variance of noise β_t, the distribution $q(z_{t-1}|z_t)$ is close to being Gaussian





EMORY

Reverse decoder

- In practice, we choose a small β_t, so that it is easier to model $q(z_{t-1}|z_t)$

- We model the reverse process using a Gaussian distribution

$$p(z_{t-1} \mid z_t, w) = N(z_{t-1} \mid \mu(z_t, w, t), \beta_t I)$$

where $\mu(z_t, w, t)$ is a deep neural network governed by a set of parameters w

- There are different ways to model $\mu(z_t, w, t)$. But now, we are going to have a lab session to use the pretrained model to generate images!